

### EXAMINER'S AMENDMENT

1. An examiner's amendment to the record appears below. Should the changes and/or additions be unacceptable to applicant, an amendment may be filed as provided by 37 CFR 1.312. To ensure consideration of such an amendment, it MUST be submitted no later than the payment of the issue fee.

Authorization for this examiner's amendment was given in a telephone interview with Mr. Mani Adeli (Reg. No. 39,585) on March 11, 2010.

2. The application has been amended as follows:

1. (Currently Amended) A computer readable medium storing a computer program for execution by at least one processor, ~~method of the computer program~~ for logging events for a plurality of applications that operate on an operating system of a first computer, the ~~method~~ computer program comprising sets of instructions for:

receiving a request from ~~an particular~~ application of said plurality of applications to create ~~an~~ event objects associated with ~~an~~ events of the application that ~~has~~ have not been logged;

in response to said request, creating the event objects in a first memory space that is separate from a second memory space in which said application operates;

from the application, receiving temporal data and other data regarding the events;

storing the received data regarding the events in the event objects; and

processing the stored data regarding the events in order to produce processed event data for display on a web browser of a second computer that is connected to the first computer through the Internet,

wherein processing the stored data further comprises processing the stored data regarding a plurality of events in order to group the stored data of the plurality of events into a plurality of hierarchical sets of events, wherein each hierarchical set comprises a parent event and at least one child event,

for a set of said hierarchical sets of events, defining a nested hierarchical display of event data for display on said web browser, said nested hierarchical display displaying the parent events at one aligned level in the hierarchy and displaying the child events at another aligned level in the hierarchy under the corresponding parent events,

wherein said receiving the request, said creating, said receiving the data, said storing, and said processing are performed by an event logging mechanism that is part of said operating system and ~~operating~~ operates independently from said application on the first computer on which said application operates.

2. (Currently Amended) The computer readable medium of A method according to claim 1, wherein the computer program further comprising comprises a set of instructions for:

checking whether event logging ~~has been~~ is turned on for the event.

3. (Currently Amended) The computer readable medium of A method according to claim 2, wherein a plurality of event objects are created for the plurality of events, wherein said creating, receiving the data, and storing are performed for each of said plurality of events having event logging turned on, wherein a plurality of event objects are created for a plurality of events.

4. (Currently Amended) The computer readable medium of A method according to claim 3 1, wherein ~~said~~ the set of instructions for processing comprises a set of instructions for analyzing said event objects after event logging is turned off.

5. (Currently Amended) The computer readable medium of A method according to claim 4, wherein the set of instructions for processing further comprises sets of instructions for ~~includes:~~

allowing user definition of ~~the~~ hierarchical levels of granularity of ~~said events of whose~~ event objects that are to be analyzed; and

allowing user definition of contexts for differentiating repeated occurrences of events deemed identical by the defined contexts ~~nature of their hierarchical position.~~

6. (Currently Amended) The computer readable medium of A method according to claim 5, wherein the set of instructions for processing further comprises sets of instructions for ~~includes:~~

grouping events into ~~their~~ hierarchical subgroups; and

grouping events by ~~their~~ context, ~~if any are defined.~~

7. (Currently Amended) The computer readable medium of A method according to claim 6, wherein the set of instructions for processing further comprises sets of instructions for:

traversing through the ~~hierarchy of~~ hierarchical subgroups until ~~the~~ a subgroup of finest granularity is traversed;

subdividing said events into further subgroups;

computing statistics for each subgroup while traversing; and

displaying said statistics.

8. (Currently Amended) The computer readable medium of A method according to claim 7, wherein the set of instructions for processing further comprises sets of instructions for if said subgroup of finest granularity has been traversed, then:

aggregating events deemed identical by virtue of ~~their~~ the events' hierarchical position into an aggregate;

computing statistics for each aggregate; and

displaying said statistics for each said aggregate.

9. (Currently Amended) The computer readable medium of A method according to claim 7, wherein ~~said~~ the set of instructions for processing further comprises sets of instructions for ~~includes~~:

aggregating events deemed identical by virtue of ~~their~~ the events' context into an aggregate;

computing statistics for each aggregate; and

displaying said statistics for each said aggregate.

10. (Currently Amended) A first computer comprising computer readable storage for storing:

an operating system comprising an event logger; and

a plurality of applications operating on said operating system,

said event logger operating independently of said applications,

said event logger for receiving input through a web browser of a second computer to enable event logging, wherein said second computer is connected to said first computer through the Internet,

wherein when said event logging is enabled, the event logger (i) identifies a set of events for a particular application ~~in the plurality of applications that generates web pages,~~  
~~responds to selections received through said generated web pages, and operates~~ ~~operating~~ on said operating system;[[,]] (ii) generates an event log;[[,]] and (iii) records event data regarding the identified set of events in the event log, said event log generated and said event data recorded in the event log without referencing any event logs of said application;

wherein the set of event data relates to at least one of a request for a web page,  
generation of the requested web page, and a request received through a selection of an item of  
the generated web page,

wherein said event ~~logging mechanism~~ logger performs the ~~receiving~~, identifying, generating, and recording on said first computer on which said particular application operates,

wherein each event in the set of events is enabled or disabled separately,

wherein when said event logging is disabled through the web browser of said  
second computer, said event logger foregoing the identifying, generating, and recording on said  
first computer on which said application operates.

11. (Currently Amended) ~~A~~ The computer ~~of~~ ~~according to~~ claim 10, wherein said event logger records the event data by recording start time, end time, and other event data into an event object for each event in the set of events ~~to be logged~~.

12. (Currently Amended) ~~A~~ The computer ~~of~~ ~~according to~~ claim 10, wherein said operating system comprises a foundational layer, wherein said event logger operates on said foundational layer.

13. (Currently Amended) ~~A~~ The computer of ~~according to~~ claim 12, wherein said foundational layer is a programmable framework.

14. (Currently Amended) ~~A computer~~ The method of ~~according to~~ claim ~~10~~ 39, ~~wherein said event logger can be enabled and then disabled beyond an execution space of said particular application operating in said operating system, wherein each event in the set of events can be enabled or disabled separately wherein said single computer is a first computer, wherein said event logging mechanism is for receiving input through a web browser of a second computer to enable event logging, wherein said second computer is connected to said first computer through the Internet.~~

15. (Currently Amended) ~~A computer~~ The method of ~~according to~~ claim ~~10~~ 14, wherein when said event logging is disabled through the web browser of said second computer, said event logger ~~foregoing foregoes~~ the creating-identifying, analyzing-generating, grouping, and defining-recording on said first computer on which said application operates.

16. (Currently Amended) ~~A~~ The computer of ~~according to~~ claim 10, wherein said event logger is further for generating a plurality of event objects and analyzing said event objects in order to generate an analyzed result for display on said web browser.

17. (Currently Amended) ~~A~~ The computer of ~~according to~~ claim 16, wherein said event logger is configured to analyze said event objects based upon hierarchical and contextual groupings.

18. (Currently Amended) ~~A~~ The computer of ~~to~~ claim ~~16~~ 17, wherein said event logger is configured to aggregate said event objects deemed identical based upon at least one of said hierarchical and contextual groupings.

19-22. (Canceled)

23. (Currently Amended) A computer comprising computer readable storage for storing:

an operating system comprising an event logger; and

a plurality of applications for operating on said operating system<sub>1</sub>[[;]]

the event logger for (i) receiving a request to create an event log, (ii)

recording event data for each of a plurality of events of ~~a~~ the plurality of applications in a first memory space that is uniquely allocated for the event logger and is separate from a second memory space allocated for the plurality of applications; ~~(ii)~~ (iii) grouping a plurality of said event data into a plurality of hierarchical sets of events, wherein each hierarchical set comprises a parent event and at least one child event[[:]], (iv) analyzing said event log according to hierarchical grouping, and (v) for a plurality of said hierarchical sets of events, defining a nested hierarchical display of event data that displays the parent events at one aligned level in the a hierarchy and displays the child events at another aligned level in the hierarchy under the corresponding parent events,

wherein said event logger is turned on, turned off, and configured using a web browser.

24. (Currently Amended) ~~A~~ The computer of ~~according to~~ claim 23, wherein said operating system further comprises a framework, said framework comprising said event logger.

25. (Canceled)

26. (Currently Amended) ~~A~~ The computer of ~~according to~~ claim 23 further comprising a first area of memory allocated to a particular application of said applications, a

second area of memory allocated to the event logger, wherein said first area of memory allocated to the particular application is separate from the second area allocated to the event logger.

27. (Currently Amended) ~~A computer~~ The method of ~~according to claim 23~~ 14, wherein said event ~~logger~~ logging mechanism ~~is can be~~ turned on, turned off, and configured using ~~a the~~ the web browser of the second computer.

28. (Currently Amended) ~~A~~ The computer ~~of~~ according to claim 23, wherein said event logger is further for allowing a user to enable and disable event logging for each event in the plurality of events, wherein said recording event data is performed for each event that has ~~having~~ event logging enabled.

29. (Currently Amended) ~~A~~ The computer ~~of~~ according to claim 23, ~~wherein the~~ operating system comprises ~~wherein the framework is~~ a framework upon which applications are executed.

30. (Currently Amended) An event logging method comprising:  
for each of a plurality of events that needs to be logged within a plurality of applications operating on an operating system of a computer:

receiving a request to create an event log;

recording event data for each of a plurality of application events in a first memory space that is uniquely allocated for ~~the event logging method~~, said first memory space separate from a second memory space allocated for the plurality of applications;

for each application in said plurality of applications, grouping a plurality of said event data into a plurality of hierarchical sets of events, wherein each hierarchical set comprises a parent event and at least one child event; and



for a plurality of said hierarchical sets of events, defining a nested hierarchical display of event data that displays the parent events at one aligned level in the hierarchy and displays the child events at another aligned level in the hierarchy under the corresponding parent events, wherein said recording, grouping, and defining are performed by an event logging mechanism that is part of the operating system of said computer and that runs independently from said applications on said computer,

wherein each of a plurality of application events comprises an enable/disable state, wherein a disable state of a particular event precludes the event logging mechanism from logging said particular event.

31. (Currently Amended)~~A The method of according to~~ claim 30, wherein the operating system comprises a framework, said framework comprising the event logging mechanism.

32. (Currently Amended)~~A The method of according to~~ claim 30 further comprising checking, for each event identified by for a particular application ~~within in~~ the plurality of applications, whether event logging ~~has been~~ is enabled.

33. (Currently Amended)~~A The method of according to~~ claim ~~32~~ 30, wherein said nested hierarchical display is for display[[ed]] in a web browser.

34. (Currently Amended)~~A The method of according to~~ claim 30, wherein the first memory space ~~occupied by the event data is within~~ a memory space that ~~has been~~ is allocated solely ~~to the~~ for event logging mechanism.

35. (Currently Amended) ~~A The method of according to~~ claim 30, wherein the events that are logged by the event logging mechanism have not been previously logged by any other application.

36. (Currently Amended) ~~A The method of according to~~ claim 30, wherein information placed in the event data is first logged by the event logging mechanism.

37. (Currently Amended) ~~A The method of according to claim 30~~ 39, wherein each of a plurality of events comprises an enable/disable state, wherein a disable state of a particular event precludes ~~any system the event logging mechanism~~ from logging said particular event.

38. (Currently Amended) ~~A The method of according to~~ claim 33, wherein said computer is a first computer, wherein said web browser executes on a second computer that is connected to said first computer through the Internet.

39. (Currently Amended) A method of logging events for an application, said method comprising:

receiving a request from the application to create an event log;

receiving a set of events generated by said application;

creating a corresponding set of event data in a first memory space that is separate from a second memory space in which the application executes;

analyzing the set of event data by determining a procedural level at which each event is executed in the application;

hierarchically grouping said set of event data into a plurality of groups based on said analysis of the set of event data, wherein each event executed at a same procedural level in the application is grouped in a same group;

defining a hierarchical display of event data comprising a nested hierarchical presentation of data for each of said events based on said grouping, said hierarchical display presented in a web browser, wherein said receiving a request, receiving the set of events, creating, analyzing, grouping, and defining—generating are performed by an event logging mechanism that is part of an operating system that runs—running independently from said application on a single computer on which said application executes.

40. (Currently Amended)—~~A~~ The method of~~according to~~ claim 39, wherein said web browser is a web browser executing on said single computer.

41. (Currently Amended)—~~A~~ The method of~~according to~~ claim 39, wherein said single computer is a first computer, wherein said web browser executes on a second computer that is connected to said first computer through the Internet—~~further comprising~~.

42. (Currently Amended) A computer comprising computer readable storage for storing:

an operating system comprising an event logger; and

a plurality of applications operating on said operating system, wherein at least one particular application is for generating web pages and responding to web page selections received through said generated web pages,

wherein the event logger is for~~[:]]~~ (i) receiving a request from the particular application to create an event log,~~(ii)~~ (ii) functioning interoperably with but separately from said applications,~~(iii)~~ (iii) identifying a plurality of event data for a plurality of application events of the particular application, said application events comprising a request for a web page, generation of a~~the~~ web page, and a request received through a selection of an item of the

generated web page, ~~(iii)~~(iv) storing the event data in ~~an application~~ the event log, (v) analyzing of said event log according to hierarchical grouping,

wherein each of said plurality of application events comprises an enable/disable state, wherein the disable state precludes the event logger from creating an event log, wherein the identifying and storing are performed for each event that has an enabled state.

43. (Currently Amended) ~~A~~ The computer of ~~according to~~ claim 42, wherein said storing said event data comprises storing, for each event ~~to be~~ that is logged, a start time, end time, and information regarding the event.

44. (Canceled)

45. (Currently Amended) ~~A~~ The computer readable medium of ~~according to~~ claim ~~42~~ 1, wherein said ~~plurality of application~~ events each comprises an enable/disable state, wherein each event that is in the disable state precludes the ~~the event logger logging mechanism~~ from creating an event object for the event log, wherein the ~~identifying creating~~ and storing are performed for each event that is having event logging in the enabled state.

46. (Currently Amended) A method of logging events for a plurality of applications that operate on an operating system of a first computer, the method comprising:

identifying a set of events that has not been logged by an application;

receiving a request from a particular application of said applications to create event objects associated with events of the application that have not been logged;

in response to said request, creating the event objects in a first memory space that is separate from a second memory space in which said application operates;

from the application, receiving temporal data and other data regarding the events;

storing the received data regarding the events in the event objects; and  
processing the stored data regarding the events in order to produce processed  
event data for display on a web browser of a second computer that is connected to the first  
computer through the Internet, analyzing the set of events identified for the application, wherein  
each event comprises at least a start time and an end time;

grouping the set of events based on said analysis of the set of events; and  
wherein processing the stored data further comprises processing the stored data  
regarding a plurality of events in order to group the stored data of the plurality of events into a  
plurality of hierarchical sets of events, wherein each hierarchical set comprises a parent event  
and at least one child event;

for a set of said hierarchical sets of events, defining a nested hierarchical display  
of event data for display on said web browser, said nested hierarchical display displaying the  
parent events at one aligned level in the hierarchy and displaying the child events at another  
aligned level in the hierarchy under the corresponding parent events,

wherein said receiving the request, said creating, said receiving the data, said  
storing, and said processing generating a display of said set of events based on said grouping,  
wherein said identifying, analyzing, grouping, and generating are performed by an event analysis  
logging mechanism that is part of said operating system and operates running on a single  
computer on which said application runs, said event analysis mechanism running independently  
from said application on said first computer on which said application operates.

47. (Currently Amended) The method of claim 46, wherein the second computer and the third computer are the same computer ~~said set of events are identified by an event logging mechanism prior to performing said analyzing, grouping, and generating.~~

48. (Currently Amended) The method of claim 46 further comprising determining whether event logging has been turned on for the event, ~~wherein said generated display is for displaying said groupings of said set of events in a web browser.~~

49. (Currently Amended) The method of claim 46, wherein said creating, receiving the data, and storing are performed for each event having event logging turned on, wherein a plurality of event objects are created for the plurality of events ~~grouping comprises grouping said set of events into a hierarchy, wherein at least one event in the hierarchy includes a sub-event.~~

50. (Currently Amended) The method of claim 46, wherein processing comprises: aggregating events deemed identical by virtue of the events' context into an aggregate; further comprising computing statistics for each of said aggregates; and using the start time and the end time of the set of events, ~~wherein said generating comprises including the statistics in the display~~

displaying said statistics for each of said aggregate.

3. Any inquiry concerning this communication or earlier communications from the examiner should be directed to DIEM K. CAO whose telephone number is (571)272-3760. The examiner can normally be reached on Monday - Friday, 7:30AM - 4:30PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Hyung Sough can be reached on (571) 272-6799. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/DIEM K CAO/  
Primary Examiner  
Art Unit 2194

DC  
March 12, 2010